On-target Rapid Prototyping using Simulink and Embedded Coder

- P. Gandhimathi
  (Electronics and Advanced Technologies/Research and Advanced Engineering)
Agenda

1. Introduction
2. Case study
3. How do we apply PIL
4. Benefits
5. Application
6. Features that can help us
Introduction

- Efficient development with simulation and verification of possible solutions in advance.

- Reduce non homogeneous behavior between simulation environment and actual control hardware.

- Target specific rapid prototyping of control system applications with MathWorks tools.

- Optimization and correction of the application on-target with simulation in Simulink.

- Lot of time and effort saving.
Case Study

- Increased computerization of modern vehicles.

- Continuous increase in number of ECUs in vehicles, with increasing safety and comfort requirements.

- Need to avoid increasing number of ECUs in vehicles.

- Fitting the new applications in the Existing ECUs.

- MathWorks tools support in making efficient process.
Fitting new application to existing ECU

- Identifying following characteristics.
  1. Interfaces
  2. Memory
  3. Periodicity

- Challenge of identifying the Memory and Periodicity needs of the new application.

- Impact of identifying memory needs at the end stage of development.

- Identifying performance characteristics in simulation environment.

- MATLAB Embedded Coder software in checking performance.
  1. Verifying the deployment object code on target processors without modifying the original model.
  2. Memory need in target processor with .map files.
  3. “Real-Time Execution Profiler” giving the timing needs on real time processor.

- Working with fixed-point code when existing ECU is a fixed-point processor.
Fixed-point conversion

- Providing low cost solution.
- Fixed point processors requiring the fixed-point code.
- Effort and error on manual fixed point calculations.
- Fixed-point calculations inside MATLAB with Simulink Fixed-point Tool.
- Easy comparison of results between floating-point and fixed-point model simulations.
- Easy debugging and tuning with data type visible at each level.
- Comparison of floating-point and fixed-point algorithm performance on target processor achieved with PIL simulation in Simulink.
Processor in loop simulation
Memory Calculation

Memory size measure

<table>
<thead>
<tr>
<th>Sub Module</th>
<th>RAM (Byte)</th>
<th>ROM (Byte)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithm 1</td>
<td>774</td>
<td>5036</td>
</tr>
<tr>
<td>Algorithm 2</td>
<td>774</td>
<td>6194</td>
</tr>
<tr>
<td>Algorithm 3</td>
<td>664</td>
<td>7352</td>
</tr>
<tr>
<td>Algorithm 4</td>
<td>770</td>
<td>8510</td>
</tr>
</tbody>
</table>

Generated .map file

Settings in MATLAB
Profiling Report

- Settings in MATLAB
- Time profile output
- Generated profiling report

<table>
<thead>
<tr>
<th>Task</th>
<th>Maximum turnaround time</th>
<th>Average turnaround time</th>
<th>Maximum execution time</th>
<th>Average execution time</th>
<th>Average sample time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run-Rate</td>
<td>0.000250 at 0.00328</td>
<td>0.00026</td>
<td>0.000251 at 0.00328</td>
<td>0.00026</td>
<td>0.0007998</td>
</tr>
<tr>
<td>Sub-Rate 1</td>
<td>0.000219 at 0.00278</td>
<td>0.000213</td>
<td>0.000202 at 0.00118</td>
<td>0.000202</td>
<td>0.0015996</td>
</tr>
<tr>
<td>Sub-Rate 2</td>
<td>0.000192 at 0.00359</td>
<td>0.00019</td>
<td>0.00019 at 0.00359</td>
<td>0.00019</td>
<td>Inf</td>
</tr>
<tr>
<td>Interrupt 3</td>
<td>1.62e-005 at 0.00822</td>
<td>1.62e-005</td>
<td>1.62e-005 at 0.00822</td>
<td>1.62e-005</td>
<td>0.0015054</td>
</tr>
<tr>
<td>Interrupt 4</td>
<td>1.62e-005 at 0.0028</td>
<td>1.62e-005</td>
<td>1.62e-005 at 0.0028</td>
<td>1.62e-005</td>
<td>0.0015907</td>
</tr>
</tbody>
</table>

Model Execution Profiling Results

- Task: Run-Rate
- Maximum turnaround time: 0.000250 at 0.00328
- Average turnaround time: 0.00026
- Maximum execution time: 0.000251 at 0.00328
- Average execution time: 0.00026
- Average sample time: 0.0007998

- Task: Sub-Rate 1
- Maximum turnaround time: 0.000219 at 0.00278
- Average turnaround time: 0.000213
- Maximum execution time: 0.000202 at 0.00118
- Average execution time: 0.000202
- Average sample time: 0.0015996

- Task: Sub-Rate 2
- Maximum turnaround time: 0.000192 at 0.00359
- Average turnaround time: 0.00019
- Maximum execution time: 0.00019 at 0.00359
- Average execution time: 0.00019
- Average sample time: Inf

- Task: Interrupt 3
- Maximum turnaround time: 1.62e-005 at 0.00822
- Average turnaround time: 1.62e-005
- Maximum execution time: 1.62e-005 at 0.00822
- Average execution time: 1.62e-005
- Average sample time: 0.0015054

- Task: Interrupt 4
- Maximum turnaround time: 1.62e-005 at 0.0028
- Average turnaround time: 1.62e-005
- Maximum execution time: 1.62e-005 at 0.0028
- Average execution time: 1.62e-005
- Average sample time: 0.0015907

RENAULT NISSAN TECHNOLOGY & BUSINESS CENTRE INDIA PRIVATE LIMITED
Reuse of test cases for PIL

Test Inputs in MATLAB

Floating point Simulink model

Result Comparison

S-Function Code Running On Processor
Validation Flow – Impact with MathWorks Tools

Without MATLAB

Requirement

Design

Application code Development

Test Cases for Design

Validation

Validation

Validation

Test Cases for Code

Test Cases for Code

Test Cases for Processor

Code change for processor specific

With MATLAB

Requirement

Simulink Model

Floating/Fixed-point model

Code generation with Embedded Coder

Validation

Test Cases

MATLAB
Working with MATLAB
Benefits

- Built-in fixed-point operations save time in simulation.
- Multiple simulations with different word length and scaling to see the simulation results before committing to hardware.
- Generates code for supported on-target rapid prototyping boards.
- Code can be executed on processors to verify behavioral performance and gather resource utilization metrics (Memory) through processor-in-the-loop and profiling techniques.
Application

- New safety regulation needs - optional features in vehicles to become mandatory.
- Cost effective and competitive approach to provide better product to our customer.
- Implementing new features in to the existing ECUs.
- TI processors such as C2000 and C6000 processors for some High speed calculation algorithms, Vision based applications respectively.
- Successful work with TI C2000 processors through MathWorks tools (Simulink, Fixed-point & Embedded Coder) for Electric vehicle applications.
- Verification, tuning and optimization of complex control system applications.
- Time, effort and cost saving.
Features from MathWorks that can help us on PIL simulation

- While continuing our work on innovative solutions to our customer, MathWorks Tools will assist us in future too.

- Expectations from MathWorks on processor in loop simulation
  1. IDE support on Microsoft Windows7 for MATLAB 2011
  2. Few more Embedded Target support for automotive applications (such as Microchip, ST)
Thank you for your attention!

Any Questions?